

## Claims

What is claimed is:

1. A method for caching in a tracing framework, comprising:  
firing a probe associated with a thread;  
evaluating a first predicate of the probe;  
caching the first predicate in a predicate cache associated with the thread, based on  
the evaluating of the first predicate and cacheability of the first predicate;  
and  
transferring control to the thread, based on the caching.
2. The method according to claim 1, wherein the evaluating comprises determining a  
boolean value of the first predicate.
3. The method according to claim 2, wherein the boolean value is true.
4. The method according to claim 3, further comprising:  
executing an action of the probe.
5. The method according to claim 2, wherein the boolean value is false.
6. The method according to claim 5, further comprising:  
determining whether the first predicate is cacheable.
7. The method according to claim 4, wherein cacheable is the first predicate referencing  
an immutable variable.
8. The method according to claim 4, wherein cacheable is the first predicate referencing  
a thread-specific variable.

9. The method according to claim 6, further comprising:
  - identifying the first predicate using a predicate cache identifier;
  - storing the predicate cache identifier with the probe as a probe cache identifier;
  - and
  - storing the predicate cache identifier in the predicate cache.
10. The method according to claim 1, wherein the transferring occurs if the first predicate is cached in the predicate cache.
11. The method according to claim 9, wherein the probe further comprises a second predicate of the probe.
12. The method according to claim 11, further comprising:
  - evaluating the second predicate of the probe.
13. The method according to claim 12, wherein the evaluating comprises determining a boolean value of the second predicate.
14. The method according to claim 13, wherein the boolean value is true.
15. The method according to claim 14, further comprising:
  - executing an action of the probe.
16. The method according to claim 13, wherein the boolean value is false.
17. The method according to claim 16, further comprising:
  - determining whether the second predicate is cacheable.
18. The method according to claim 17, further comprising:
  - identifying the second predicate using the predicate cache identifier, if the first and second predicate are the same.

19. The method according to claim 18, wherein cacheable is the first predicate referencing an immutable variable and the first predicate and the second predicate having the same identifier.
20. The method according to claim 18, wherein cacheable is the first predicate referencing a thread-specific variable and the first predicate and the second predicate have the same identifier.
21. The method according to claim 1, further comprising:
  - determining whether the first predicate is cached; and
  - determining whether the predicate cache is valid.
22. The method according to claim 21, wherein the determining whether the predicate is cached comprises comparing whether the probe cache identifier and the predicate cache identifier stored in the predicate cache are equivalent.
23. The method according to claim 21, wherein the determining whether the predicate cache identifier is valid comprises comparing whether the probe cache identifier and the predicate cache identifier stored in the predicate cache are non-zero.
24. The method according to claim 1, further comprising:
  - invalidating the predicate cache.
25. The method according to claim 24, wherein the invalidating comprises setting the predicate cache to zero.
26. The method according to claim 24, where in the invalidating is a result of a thread-specific variable being stored.
27. The method according to claim 1, further comprising:
  - setting the predicate cache to zero initially.

28. A computer system for caching in a tracing framework comprising:
- a processor;
  - a memory;
  - a storage device; and
  - software instructions stored in the memory for enabling the computer system to:
    - fire a probe associated with a thread;
    - evaluate a first predicate of the probe;
    - cache the first predicate in a predicate cache associated with the thread,
      - based on the evaluating of the first predicate and cacheability of the first predicate; and
    - transfer control to the thread, based on the caching.